

Multicast Heuristic Approaches on Multi-Layer Wireless Network

Mauro Tropea¹, Amilcare Francesco Santamaria²

DEIS Department, University of Calabria
via P. Bucci cubo 42/c, Arcavacata di Rende 87036 – Cosenza - Italy
¹mtropea@deis.unical.it; ²afsantamaria @deis.unical.it

Abstract

Multicast services are increased in exponential manner in these last few years and in particular in the field of multimedia services. Multicast can reduce resources allocation and enhance network performances in terms of QoS, especially in a wireless platforms where the bandwidth is a precious resource. The problem of multicast routing can be reduced to the problem of finding a spanning tree capable to distribute network flow among multicast sources and destinations. It has been established that determining an optimal multicast tree for a static multicast group can be modeled as the Steiner Tree problem in networking, this problem has been proofed to be a NP-complete. Hence the necessity of using scalable algorithms in scalable networks composed of multi layered platforms. Moreover, in this work a QoS multi-constraint multicast problem has been addressed. In this paper a comparison between two meta-heuristic algorithms is presented in order to show the scalability introduced by these types of algorithms that are able of finding sub-optimal solutions. These meta-heuristics are based on Genetic Algorithm and Simulated Annealing mechanism. A simulated campaigns between two proposed algorithm has been addressed.

Keywords

Scalable Algorithm; Multicast routing; heterogeneous platforms; Genetic Algorithm; Simulated Annealing

Introduction

This The request of multicast services are increased in exponential manner in these last few years and in particular in the field of multimedia services. Multicast can reduce resources allocation and enhance network performance in terms of QoS and resources allocation. Key factor of the multicast is the capacity to reduce packets number that flows on the network. Only necessary packets are sent by the source of a generic multicast group, after that the network will provide to send spread towards all destinations data flow. In the multicast routing two entities work together with the

main goal to distribute data among all nodes that belong to a multicast groups, these entities are the multicast protocol and the multicast algorithm, this last one can be completely disconnected from protocols, in this case the protocol has the task to trigger algorithm when needed. Use integrated or independent algorithm depends of network data to be distributed or from application type, moreover, several approaches exist to implements multicast such as centralized or distributed, shared or not, static or dynamic and so on. With rapidly growth of hardware technologies and the rapid evolution of Internet even more kind of applications was born, that require QoS constraints. Multicast routes its packets following a multicast tree that is searched or built in order to reach each destination. The problem of multicast routing can be reduced to the problem of finding a tree spanning the source and all destinations that belong to the multicast group. It has been established that determining an optimal multicast tree for a static multicast group can be modeled as the Steiner tree problem in networks, which is proofed to be a NP-complete problem. Hence the necessity of using scalable algorithms in scalable networks composed of multi layered platforms [1][2], in particular a multi-constraints QoS multicast problem is faced. In order to face the scalable issues also mechanisms of Call Admission Control can be previewed for better exploiting wireless resources [3]. Many mechanisms exist that try to make the multicast algorithms scalable. They are heuristic search whose purpose is to find a sub-optimal solution. Some mimics the process of natural evolution as Genetic Algorithms; others are based on the collective behavior of decentralized, self-organized systems, known as Swarm Intelligence; others are inspired on the metallurgic process called annealing that is a technique involving heating and controlled cooling of a material to increase the size of its crystals and reduce their defects. In this paper a comparison between two meta-heuristic algorithms is

presented in order to show the scalability introduced by these types of algorithms that are able of finding a sub-optimal solution into a solution space. These meta-heuristics are based on Genetic Algorithm (GA) and Simulated Annealing (SA) mechanism. A simulated campaigns of a comparison between the two proposed algorithm are shown. The characteristics of both algorithms are highlighted. The remainder of this paper is organized as follows. The related work is briefly described in section II. The Multi-Layer reference scenario is presented in section III. Section IV describes the multicast QoS algorithms, Genetic Algorithm and Simulated Annealing Algorithm. Simulation results, showing the algorithms comparison, are summarized in section VI and finally conclusions are presented in section VII.

Related Work

The issues of building a multicast tree through multicast algorithms that guarantee multi-constraints have been carried out to resolve the multi-constraints QoS and various approaches have been proposed. In the following some works on multicast algorithms are presented. In [4] a new heuristic algorithm is presented for constructing minimum-cost multicast trees with delay constraints. The new algorithm can set variable delay bounds on destinations and handles two variants of the network cost optimization goal: one minimizing the total cost (total bandwidth utilization) of the tree, and another minimizing the maximal link cost (the most congested link). Instead of the single-pass tree construction approach used in most previous heuristics, the new algorithm is based on a feasible search optimization method which starts with the minimum-delay tree and monotonically decreases the cost by iterative improvement of the delay-bounded tree. The optimality of the costs of the delay-bounded trees obtained with the new algorithm is analyzed by simulation. Depending on how tight the delay bounds are, the costs of the multicast trees obtained with the new algorithm are shown to be very close to the costs of the trees obtained by the Kou, Markowsky and Berman's algorithm (1981). In [5], authors propose a novel QoS-based multicast routing algorithm based on the genetic algorithms (GA). In the proposed method, Prüfer number is used for genotype representation. Some novel heuristic algorithms are also proposed for mutation, crossover, and creation of random individuals. They evaluate the performance

and efficiency of the proposed GA-based algorithm in comparison with other existing heuristic and GA-based algorithms by the result of simulation. This proposed algorithm has overcome all of the previous algorithms in the literatures. In [6], it is formulated as a multiobjective constrained combinatorial optimization problem for a protocol to determine multicast routes satisfying different QoS requirements. Authors use a multiobjective model and routing approach based on genetic algorithm that optimizes multiple QoS parameters simultaneously. The corresponding results demonstrate that the proposed approach is capable of discovering a set of nondominated routes within a finite evolutionary generation. Its feasibility and performance has been verified. Paper in [7] deals with the problem of delay-constrained multicast routing (DCMR). The DCMR problem can be reduced to the constrained minimum Steiner tree problem in graphs (CMStTG). Since the minimum Steiner tree problem in graphs (MStTG) has been proven to be NP-complete, several heuristics have been developed for solving MStTG and CMStTG. In this paper, authors suggest a tabu search heuristic for the DCMR problem. This heuristic was developed on the basis of a tabu search heuristic designed for solving unconstrained minimum Steiner tree problems. Preliminary testing on data from a publicly available library, SteinLib, has shown that this heuristic gives near optimal solutions in moderate time and a moderate number of iterations for medium sized problems (50-100 nodes). Comparing with a well known algorithm for solving the CMStTG problem, tests have shown that our tabu search heuristic is superior in quality for medium sized problems. Swarm intelligence refers to complex behaviors that arise from very simple individual behaviors and interactions, which is often observed in nature, especially among social insects such as ants. Although each individual (an ant) has little intelligence and simply follows basic rules using local information obtained from the environment, such as ant's pheromone trail laying and following behavior, globally optimized behaviors, such as finding a shortest path, emerge when they work collectively as a group. In [8], authors apply the biologically inspired metaphor to the multicast routing problem in mobile ad hoc networks. Their proposed multicast protocol adapts a core-based approach which establishes multicast connectivity among members through a

designated node (core). An initial multicast connection can be rapidly setup by having the core flood the network with an announcement so that nodes on the reverse paths to the core will be requested by group members to serve as forwarding nodes. In addition, each member who is not the core periodically deploys a small packet that behaves like an ant to opportunistically explore different paths to the core. This exploration mechanism enables the protocol to discover new forwarding nodes that yield lower total forwarding costs, where cost is abstract and can be used to represent any metric to suit the application. Simulations have been conducted to demonstrate the performance of the proposed approach and to compare it with certain existing multicast protocols.

Reference Scenario

The meta-heuristics algorithms comparison has been conducted over a multi-layer wireless system composed of a next-generation GEO DVB-RCS Satellite and a set of HAPs. Both wireless platforms are equipped with a DVB-RCS On Board Processing (OBP) payload. The reference architecture considered in this work is depicted in Fig. 1.

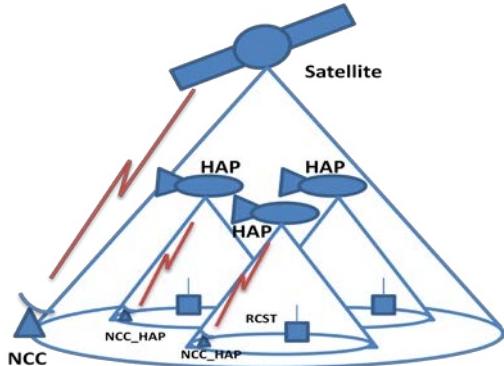


FIG. 1 REFERENCE SCENARIO

DVB-RCS system was specified by an ad-hoc ETSI technical group founded in 1999 [9]. It uses typical frequency bands in Ku (12-18 GHz) for the forward link and/or Ka (18-30 GHz) for the return link, and comprises of Return Channel Satellite Terminals (RCSTs), a Network Control Center (NCC) that is the core of the system with the task of managing the system. In particular, it manages the connection, the system synchronization and informs all the RCST about the system. RCST consists of four different types on the basis of different bandwidth capacity: RCST A (144 Kbps), B (384 Kbps), C (1024 Kbps), D (2048 Kbps). The transmission capability uses a Multi-Frequency

Time Division Multiple Access (MF-TDMA) scheme to share the capacity available for transmission by user terminals. The mapping between source traffic type and capacity category depends on the types of provided service, the used transmission protocols and constraints imposed by the satellite orbit. Capacity categories supported by the standard are Continuous Rate Assignment (CRA); Rate Based Dynamic Capacity (RBDC); Volume Based Dynamic Capacity (VBDC); Absolute Volume Based Dynamic Capacity (AVBDC). The standard previews a frame structure of the duration of 47 ms and a possible use of IP packets carried via DVB/MPEG2-TS (Transport Stream). A frame consists of a number of time slots on a certain number of carriers. The number and composition of time slots per frame is determined by the information bit rate to be supported by the frame. The frame structure consists of 188 bytes (a 4-byte header and a 184-byte payload). More details about the frame composition can be found in [9].

Multicast QoS Algorithms

In this section the multicast routing issues related to the algorithms tasks has been faced, taking into account that a telecommunication network can be viewed as a not oriented and weighted graph $G(V,E)$, where V represents network nodes set and E links set, connection between nodes. Each node represents a router on the network that allows communication among other nodes and between source and destinations. Each link has a weight between two adjacent nodes; weights represent the QoS metrics that the algorithm uses to find multicast tree. Graph (G) contains $n = |V|$ nodes and $l = |E|$ links. Each node is simply identified as $v_i \in V$ $|1 \leq i \leq n$, moreover, each link can be written as follows:

$$e_{i,j} \in E \mid i \in V, j \in V, 1 \leq i \leq n, 1 \leq j \leq n, i \neq j$$

Moreover, each link in the network has the following triple QoS metrics associated with it $(c_{i,j}, \delta_{i,j}, b_{i,j})$. In particular $c_{i,j}$ is the cost between the nodes i and j , $\delta_{i,j}$ is the delay between the nodes i and j and $b_{i,j}$ is the available bandwidth between the nodes i and j . For more details see [10]. The multicast paradigm that is considered is the one-to-many, which presumes a data distribution from a single source to several destinations. The main goal is to find a multicast tree capable to distribute packets data on the network

respecting some QoS constraints. As just declared the multicast tree searches in a multi constraints environment is well known as a NP-complete problem, these kinds of trees are also called Steiner Tree Problem (STP), therefore a STP has to be considered. For this reason is not possible to apply polynomial algorithms which are commonly used in the networking and in combinatorial field. Scalable meta-heuristic approaches have been analyzed in order to find sub-optimal solutions capable to satisfy multicast tree requirements such as QoS constraints. Two different algorithms are compared, a GA called Constraints Cost Delay Bandwidth - Genetic Algorithm (CCDB-GA) shown in 0 and a new meta-heuristics algorithm derived from metallurgic fields called Simulated Annealing (SA) [10].

Genetic Algorithm

The meta-heuristics GA [13][14] permits to find a solution that is contained into the solution space. Multicast trees are coded into one dimensional data structure where each node is represented by a chromosome, also called individual. The chromosome has a fitness function defined using the Penalty technique [11][12]. GAs are used to solve optimization problem based on evolution principle. GAs use three kind of basic operations that are Selection, Crossover and Mutation. The fitness function considered in this work is based on cost, delay and bandwidth constraints. It is represented by the following equation:

$$F(T(s, R)) = \sum_{e \in T(s, R)} \frac{a}{C(e)} \left(\prod_{d \in M} \phi(D(P(s, d)) - MD) \right) \left(\prod_{d \in M} \phi(MB - B(P(s, d))) \right) \quad (1)$$

$$\phi(z) = \begin{cases} 1 & z \leq 0 \\ \gamma & z > 0 \end{cases} \quad (2)$$

The proposed GA assigns the fitness value considering the penalty technique. A chromosome that does not respect some of the imposed bounds is penalized by the formula (2). The formula shown in (1) considers all three metrics (cost, delay and bandwidth). The first part considers the cost and the end-to-end delay, the second part concerns the bandwidth constraints. The selection probability of a parent i , P_i , is given by equation (3):

$$P_i = \frac{F(T_i)}{\sum_{j=1}^{pop_size} F(T_j)} \quad (3)$$

where $F(T_i)$ is the fitness of the T_i individual and the pop_size is the size of population. The scope of the

genetic operators is to explore the greatest portion of the space of the solutions without increasing the complexity. These are used to create a new generation starting from an initial population. These techniques do not assure that the found solution is the optimum, but with an opportune tuning phase is possible to identify those solutions that are closer as possible to the optimum. In Fig. 2. is shown as Genetic Algorithm works. In particular, it starts from an initial population and once each individual has been evaluated a finite loop is started. In this loop the population evolves in order to find a solution. As further described in this loop the genetic operators make their job, moreover, these operators permit to converge and avoiding local optimum traps exploring the solution space. Better is the tuning phase and better these operators look for a better solution. For a better explanation of GA see [11][12].

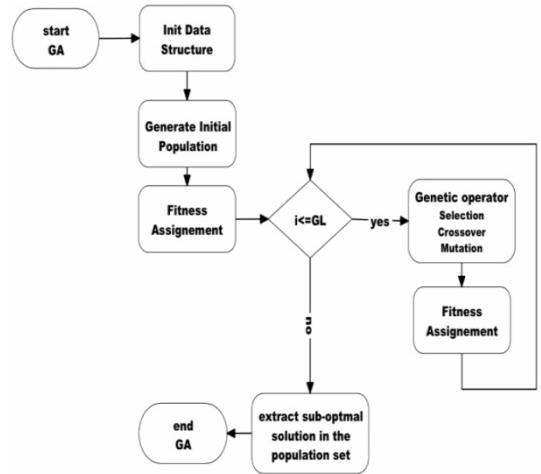


FIG. 2 GENETIC ALGORITHM FLOW CHART

Simulated Annealing

SA is a newest meta-heuristic that is carried out from the metallurgic field. In particular, it exploits the law of metals that solidify in ordered or chaotic structure following the temperature trends [10]. It is based on three steps: initialization of the parameters; generate several solutions that are neighbor of the initial solution. The changes are made taking into account temperature values; compare previous solution from solutions set with the current. If the previous solution is better than the current, this solution is taken as current. There is another possibility that the solution is taken as current, based on the uphill law. The probability depends of the temperature value. The inner energy function is used by the algorithm to make solution comparison in order to carry out the goodness solutions. This function takes into account

end-to-end delay, minimum available bandwidth and cost factor. The end-to-end delay contribution is given by:

$$\gamma = 1 - \frac{\phi(T)}{\text{DelayThreshold}} \quad (4)$$

Where $\phi(T)$ is the maximum delay among source and destination and DelayThreshold is the maximum delay that a generic packets that belongs to the multicast group can accumulate during its journey between source and destination. In order to give more details about bandwidth contribute the following formulas have to be given:

$$\rho = \min \{ \text{Bandwidth}(i, j) \} \forall e(i, j) \in E_T \quad (5)$$

$$\varphi = \max \{ \text{Bandwidth}(i, j) \} \forall e(i, j) \in E_T \quad (6)$$

Moreover with the B is indicated the transmission bandwidth, at the end the bandwidth contribute is given by the (7):

$$\begin{cases} \eta = 1 & \rho = \varphi = 0 \\ \frac{\rho - B}{\varphi} & \text{otherwise} \end{cases} \quad (7)$$

Function to evaluate the goodness of a solution is given by the (8):

$$\begin{cases} k_1 * \gamma + k_2 * \eta \\ k_1 + k_2 = 1 \end{cases} \quad (8)$$

In order to have different behaviour in terms of bandwidth and delay it is important to tune the k_1 and k_2 coefficients. In accordance with the (8) higher is k_1 higher importance is given to the end-to-end delay requirements. Inverse case is made when k_2 is higher than k_1 , in this case more importance is given to the bandwidth requirement. The temperature value influences the number of element that at each iteration could be modified. Commonly a linear temperature law is utilized and it is given by the (9):

$$t_{k+1} = \alpha * t_k \quad (9)$$

In (9) the term k is the k -th iteration and α is a real number that belongs to the range $[0..1]$. This function does not take into account dynamic evolving of the search process. A more realistic function is the proposed temperature law which takes into account the dynamic of the search process and adapts the temperature to the state of the current solution. A window mechanism is used to control temperature trend. The temperature law is shown in (10):

$$t(k) = \text{InitTemp} + \left(\frac{1}{e^k} \right) - \text{Norm} * e^{(k - \text{ItNum})} - \beta * k \quad (10)$$

Here the following term means : k is the iteration number; InitTemp is the initial temperature and it is

the starting point for the algorithm; Norm is a coefficient used to normalize values, and it permits that at the end of the iterations the temperature is around the zero; ItNum is the max number of iteration of the algorithm; β is the coefficient that it is used to give a slope to the trend of the function. Therefore, it is used to control the decreasing speed of the temperature. This coefficient allows us to control the temperature trend increasing or decreasing temperature speed. This parameter is controlled by the algorithm taking into account past values of the temperature that belong to a time window that is composed of 5 time samples of objective function. For more details on simulated annealing see [14].

TABLE 1 TOPOLOGY AND ALGORITHM PARAMETERS

Topology Parameters	Value
Number of Satellite	1
Satellite Round Trip Time (ms)	540
Number of Hap	10
Hap Round Trip Time (ms)	0.4
Hap and Satellite Medium Access Protocol	MF-TDMA
Return and Forward Channel Trama (ms)	47
Number of RCST	53
Number of Source per multicast group	1
Algorithms Parameters	Value
Mutation Probability (CCDB-GA)	0.5
Crossover Probability (CCDB-GA)	0.5
Generation Number (CCDB-GA)	100
Initial Population (CCDB-GA)	50
Fitness Penalty (γ) (CCDB-GA)	0.5
Number of Iterations	100
Initial Temperature (SA)	100
Time window size (SA)	5
Starting Solution Algorithm (SA)	Prim
K1 (SA)	0.5
K2 (SA)	0.5
Delay Bound (DB) requested (ms)	150
Bandwidth value (kbps)	1000

Multicast Algorithms Simulation Campaigns

In this section some simulative campaigns are carried out. In this works both GA and SA have been compared. In particular several simulative campaigns have been made to verify the goodness of the solutions found by the SA algorithm. The SA has been matched with the just proposed algorithm CCDB-GA that belongs to the GA family, its goodness has been proofed in others works where it outperforms other GAs, in particular in the considered multicast scenarios [11][12]. A comparison in terms of computational time has been also carried out to verify the time resources that the algorithms use to find solution. In order to better illustrate the SA contribute on Temperature control a two different simulative

campaigns are carried out. The first one take into account the Temperature Constant decreasing and its results have been compared with the GA. In the second campaigns, instead, the new temperature control is considered and SA algorithm solution have been compared with the GA solutions. In this campaign both algorithms have to face with a scenario where the network nodes increase, the main goal of this campaign is to carry out a behavior trend of the algorithms when the network size increases. In particular, it is important to observe if algorithms converge or diverge in terms of solutions in order to achieve an evaluation key in terms of goodness. The bounds on average end-to-end delay and minimum available guaranteed bandwidth are 150 ms and 1000 Kbps respectively. Looking at Fig. 3 SA algorithm supplies a higher available bandwidth than the GA, this happens because during solutions generation SA algorithm works on neighbors to find a feasible solution; In particular considering the reference network, satellite nodes give the possibility to reach other nodes with a single hop, and for this reason it is often chosen by the algorithm to cover a wide number of nodes, this allows solution to be feasible due to the coverage of multicast group and the respect of the constraints. In this way, the algorithm finds solutions where the average end-to-end delay of the multicast group is higher than the average end-to-end delay that the GA. GA can found better solutions because it investigates a wide number of feasible solutions during each iteration, see Fig. 4 for End-To-End delay Trend. Moreover, another limit of the SA, in this configuration, is that it cannot investigate a wider area of the solution space because it is limited by the temperature law that, sometimes, decreases too fast. In fact, when a feasible solution is found the SA search for solutions that are even closer to the current solution and this cannot permit us to investigate new solutions. In other words it is possible to remain trapped in a local optimum cage, because when the temperature is too low the uphill phase could be not sufficient to avoid this issue. In the second campaign the SA algorithm that implements the Dynamic Temperature mechanism is compared with the GA. In order to demonstrate better performances of the SA with Dynamic Temperature, several simulative campaigns have been carried out. In Fig. 5 the trend of the average End-To-End delay at the increasing of the multicast groups size is shown. Even in Fig. 5 it is possible to note that the GA and SA initially start with closer solutions in terms of average End-To-End delay.

Moreover, with the increase of the size of the multicast group even better solution is found because a higher region of the solution space has been investigated. This is possible because a higher number of multicast paths are available thanks to a better knowledge of the networks that has been brought by the multicast protocol messages [11][12]. In Fig. 6 the distance degree is achieved as shown in [11][12]. As shown the GA outperforms the current proposed SA algorithm in terms of goodness of found solution because it is closer to the optimum solution than the solution found by the SA. Moreover, when the multicast group size is composed of a lower number of members then the distance between the found solution and the optimum solution is greater than the distance found in case of greater multicast groups. The algorithms bring up this behavior because the unknown region of network topology does not allow algorithms to search a solution in all region of the solution space, therefore all possible paths cannot be investigated. In order to perform a computational time comparison a constant number of iterations for both algorithms is considered. The number of iterations has been carried out by several simulative campaigns that allow algorithms to expire transitory time, thus the better sub-optimal solution is found by the algorithms. In particular, this is made to verify how the algorithms perform the search and how many times they require to achieve solution. In Fig. 7 the computational time is shown. It is possible to observe that computational time increases with the increase of the network size and here the GA outperforms the SA, therefore observing the trend of the SA algorithm is possible to affirm that more enhancements are needed to achieve better results in order to reach the goodness and the performances of the GA. In particular, it will be possible to work on the neighbor and uphill procedures to achieve better results in terms of computational time. Moreover, a more better temperature function shall be carried out to outperforms current version of the SA algorithm. In Fig. 8 the minimum available bandwidth along the paths between the source of the multicast group and each destinations is shown. In this case the SA outperforms the GA when the network size is composed of a lower number of network nodes, instead when the network size increases then the goodness of the solution in terms of available bandwidth are closest. Moreover, when the network size increases then higher number of branches that are involved into solution increase. This augments the

probability to found a branch that have a reduced bandwidth availability. Therefore, these branches limit the available bandwidth in resource reservation process.

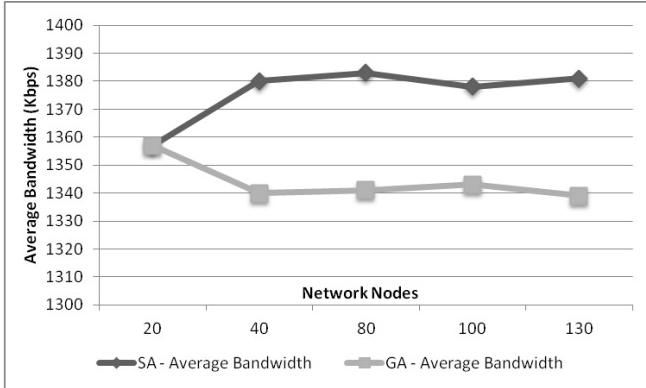


FIG. 3 AVERAGE BANDWIDTH VERSUS NETWORK NODES

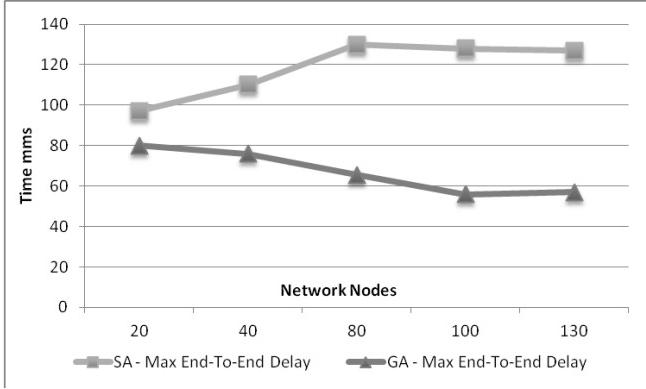


FIG. 4 MAX END-TO-END DELAY VERSUS NETWORK NODES

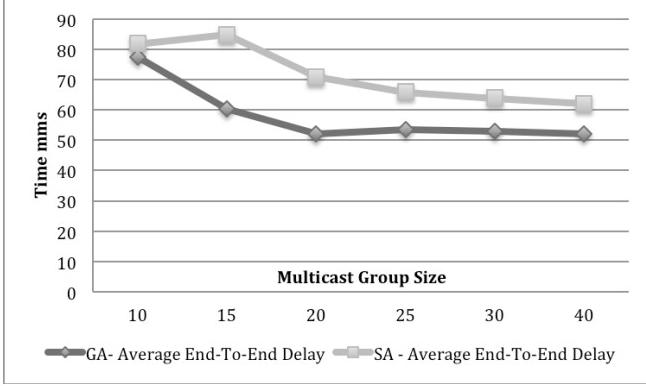


FIG. 5 AVERAGE END-TO-END DELAY RESULTS COMPARISON

Conclusions

In this works several simulative campaigns have been carried out to verify the goodness of the purposed SA algorithm and to make a comparison between SA algorithm and a GA one. The proposed SA algorithm has good performances when a limited network size is considered. Instead with huge network the SA presents some limits in terms of computational time and goodness of found solutions. As shown in the

Simulative campaigns the Dynamic temperature, which has been proposed in this works, outperforms the classic and constant decreasing of the temperature. The proposed mechanism permits to achieve results that are much closer to the CCDB-GA than the classic approach and this permits to consider the SA as a valid alternative to the GAs. Therefore, more works are needed in order to enhance the algorithm performances also in those cases where the GA outperforms the SA. In particular, the uphill process that allows us to explore a higher solution space area has to be enhanced to perform a better search. In order to reduce computational time shall be possible to reduce time in neighbor process generation. In order to find a more feasible solution shall be possible to enhance the Inner Energy Function, considering an auto-adaptive window mechanism and temperature control management to outperform current one. This work shows that both algorithms offer good performances, but both require a setup phase where the algorithm parameters can be well configured. In this way the algorithm can perform a better solution space exploration. In next works the enhancement on the GA algorithm and the simulation results varying the algorithm parameters will be shown .

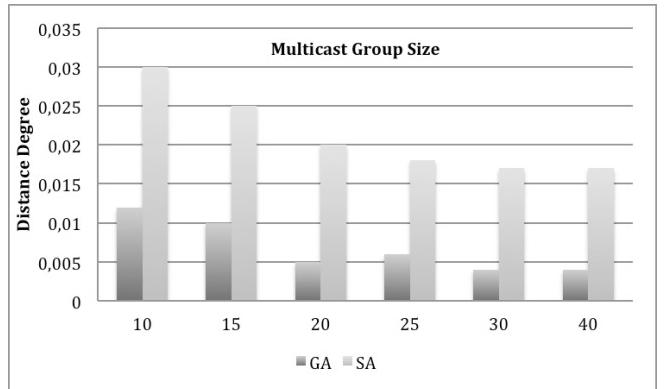


FIG. 6 SOLUTION DISTANCE DEGREE

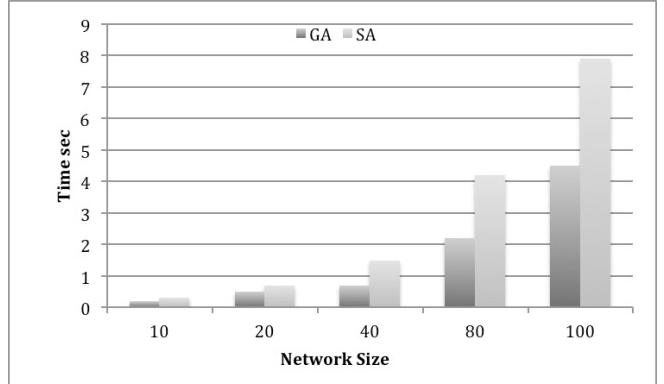


FIG. 7 COMPUTATIONAL TIME COMPARISON

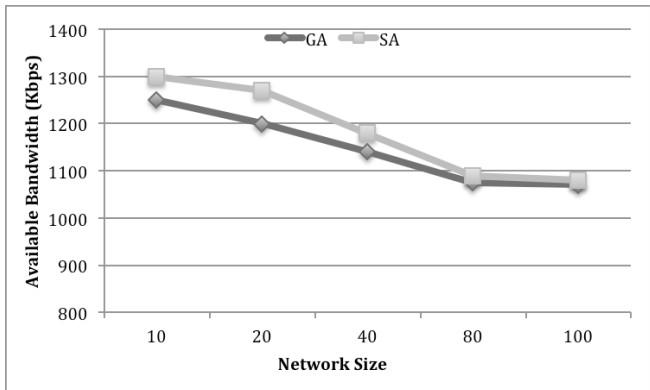


FIG. 8 AVAILABLE BANDWIDTH ALONG MULTICAST TREE BRANCHES

REFERENCES

- [1] De Rango F., Tropea M., Marano S., "Integrated Services on High Altitude Platform: Receiver Driven Smart Selection of HAP-Geo SatelliteWireless Access Segment and Performance Evaluation," International Journal Of Wireless Information Networks, 2006.
- [2] Molinaro A., De Rango F., Marano S., Tropea M., "A scalable framework for end-to-end QoS assurance in IP-oriented terrestrial-satellite networks". IEEE Communications Magazine, Vol. 43, pp. 130-137, Apr. 2005.
- [3] De Rango F., Tropea M., Fazio P., Marano S., "Call Admission Control for Aggregate MPEG-2 Traffic over Multimedia Geo-Satellite Networks," to be published on IEEE Transaction on Broadcasting, 2009.
- [4] Zhu Q., Parsa M., Garcia-Luna-Aceves J.J., "A Source-Based Algorithm for Delay-Constrained Minimum-Cost Multicast", 1995 IEEE, pp.337-385.
- [5] Haghigat A.T., Faez K., Dehghan M., Mowlaei A., Y.Ghahremani, "A Genetic Algorithm for Steiner Tree Optimization with Multiple Constraints Using Prüfer Number", Proc. of the 15th Int. Conf. on Computer Communication, 2002.
- [6] Cui X., Lin Chuang, Wei Yaya, "A multiobjective Model for QoS Multicast Routing based on Genetic Algorithm", proc of ICCNMC'03
- [7] Skorin-Kapov, Kos M., "The Application of Steiner Trees to Delay Constrained Multicast Routing: a Tabu Search Approach. 7th Internat. Conference on Telecommunications – ConTel 2003, June 11-13, Zagreb, Croatia
- [8] Chien-Chung Shen, Chaiporn Jaikaeo, "Ad hoc Multicast Routing Algorithm with Swarm Intelligence", Mobile Networks and Applications, Volume 10, Numbers 1-2 / February, 2005
- [9] ETSI, "Interaction channel for Satellite Distribution Systems" ETSI EN 301 790 V1.5.1 – 2009-05.
- [10] De Rango F., Santamaria A.F., Tropea M., Marano S., "Meta-heuristics methods for a NP-Complete networking problem," 66th IEEE Vehicular Technology Conference (VTC Fall 2008), Calgary, Alberta, Canada, 21-24 Sept. 2008.
- [11] De Rango F., Tropea M., Santamaria A.F., Marano S., "Multicast QoS Core-Based Tree Routing Protocol and Genetic Algorithm Over an HAP-Satellite Architecture", IEEE Transaction on Vehicular Technology, Oct.2009.
- [12] De Rango F., Tropea M., Santamaria A.F., Marano S., "An Enhanced QoS CBT Multicast Routing Protocol based on Genetic Algorithm in a Hybrid HAP-Satellite System", Computer Communication Journal (Elsevier), pp. 1-21, Dec.2007
- [13] Chen L., Yang Z.Y., Xu Z.Q., "A Degree-Delay-Constrained Genetic Algorithm for Multicast Routing Tree," Proc. of 4th Int.Conf on Comp. and Inf. Technology (CIT'04), 2004.
- [14] Feng Y.L., Yu Z.W., Pan Y., "Heuristic Genetic Algorithm for Degree-Constrained Multicast Routing Problem," Proc. of 3rd Int.Conf.on Machne Learning and Cybernetics, Shanghai, 26-29 Aug, 2004.

Mauro Tropea graduated in computer engineering at the University of Calabria, Italy, in 2003. Since 2003 he has been with the telecommunications research group of D.E.I.S. in the University of Calabria. In 2004 he won a regional scholarship on Satellite and Terrestrial broadband digital telecommunication systems. Since November 2005 he has a Ph.D student in Electronics and Communications Engineering at University of Calabria. His research interests include satellite communication networks, QoS architectures and interworking wireless and wired networks, mobility model.

Amilcare-Francesco Santamaria graduated in computer engineering at the University of Calabria, Italy, in 2005. Since 2005 he has been with the telecommunications research group of D.E.I.S. in the University of Calabria. His research interests include satellite communication networks, HAPs networks, Hybrid networks, Genetic Algorithm, Swarm Intelligence, QoS multicast routing, mobility model.